**PATENT APPLICATION
DOCKET NO. 10007687-1**

**IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE**

INVENTOR(S): Shell S. Simpson   **CONFIRMATION NO:** 1962

SERIAL NO.: 09/923,328   **GROUP ART UNIT:** 2143

FILED: August 8, 2001   **EXAMINER:** Bilgrami, Asghar H.

SUBJECT: METHOD FOR OBTAINING STATUS OF WEB-BASED IMAGING
ORIGINATED PROCESSES

———————————————

APPELLANTS'/APPLICANTS' REPLY BRIEF

The Appellant filed an opening brief on March 21, 2007. The Examiner mailed an answer addressing the opening brief on July 17, 2007. The following is a reply to the Examiner's answer.

1.   GROUNDS FOR REJECTION TO BE REVIEWED.

    A.   Claims 1-37 were rejected under 35 U.S.C. §103 as being unpatentable over US. Pub 2002/0138564 to Treptow in view of US Pub 2005/0228711 to Lahey.

2.   ARGUMENT.

    A.   Ground For Rejection A – Claims 1-37 were rejected under 35 U.S.C. §103 as being unpatentable over US. Pub 2002/0138564 to Treptow in view of US Pub 2005/0228711 to Lahey.

   **Claim 1** is directed to a method for monitoring a web-based service and recites the following acts:

1. receiving automatically at a client a service reference to a status of a job in a network service;
2. adding the service reference to a bookmark list on the client; and
3. removing automatically the service reference from the bookmark list on the client when the job is completed by the network service.

    In the opening brief, the Appellant explained that Treptow and Lahey failed to teach or suggest receiving automatically at a client a service reference to a status of a job in a network service or adding the service reference to a bookmark list on the client. It follows

also then that Treptow fails to teach or suggest removing a reference from a bookmark list. Treptow does teach the use of a client to display a web page through which the user can discern the job status of various job requests. Logically, the client in some fashion receives an URL or some other reference for that web page. However, Treptow makes no indication that the URL for the web page is received automatically at a client or added to a bookmark list on that client.

Attention is drawn to paragraph [0181] which states that a book mark list is a bookmark list provided in a browser. In other words, a book mark list is a browser feature that can be configured by a user. A user can add bookmarks to the list, so that when the user accesses the browser at a later time, the user can select the added bookmark using the browser's bookmark feature. In this fashion, the user does not need to remember complicated references or URL's for websites. Instead, the user simply adds bookmarks to a bookmark list managed by a web browser.

Addressing the Appellant's explanation on page 8 of the answer, the Examiner equates the information displayed in a web page illustrated in Treptow's Fig. 13, shown below. A review of the illustration reveals an administrative panel web page (292) displayed by a browser. The web page (292) includes list of status information corresponding to a series of jobs. If one takes a closer look at Fig. 13, it is evident that the web page 292 is being displayed by a browser, in this case Internet Explorer. Across the top of Fig. 13, various labels corresponding to features of the browser are displayed. Those labels include FILE, EDIT, VIEW, FAVORITES, TOOLS, and HELP. The label FAVORITES corresponds to the bookmark feature of the browser. That feature is used to manage a bookmark list for the browser. Treptow mentions nothing of adding any reference or other piece of information displayed by web page 292 to a bookmark list for the browser. Treptow also fails to teach the removal of such references from a bookmark list.
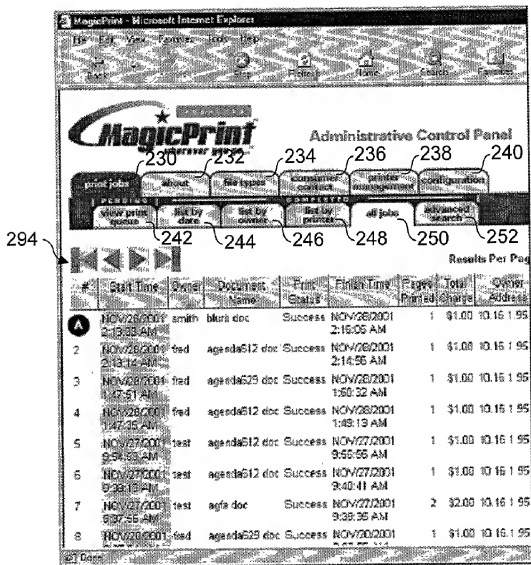
FIG. 13

For at least these reasons, Claim 1 is patentable over the cited references as are Claims 2-4, which depend from Claim 1.

Claims 5, 14, 18, and 19 are independent claims that, like Claim 1, recite acts, or system elements for implementing acts, in which a service reference is automatically received and added to a bookmarks list. As clarified above, the cited references do not teach or suggest such acts. For the same reasons Claim 1 is patentable over the cited references so are Claims 5, 14, 18 and 19. Claims 6-13 depend from Claim 5 while Claims 15-17 depend from Claim 14 and are each patentable due their dependence from a patentable base claim.

Claim 20 is directed to a program product that includes machine readable program code for causing a machine to perform the following method of Claim 1. For at least the same reasons Claim 1 is patentable, so are Claim 20 and Claims 21-23 which depend from Claim 20.

Claim 24 is directed to a program product that includes computer readable program code, that when executed, implements the method of Claim 5. For at least the same reasons Claim 5 is patentable, so are Claim 24 and Claims 25-32 which depend from Claim 24.

Claim 33 is directed to a program product that includes computer readable program code, that when executed, implements the method of Claim 14. For at least the same reasons Claim 14 is patentable, so are Claim 33 and Claims 34-36 which depend from Claim 33.

Claim 37 is directed to a program product that includes computer readable program code, that when executed, implements the method of Claim 18. For at least the same reasons Claim 18 is patentable, so is Claim 37.

For at least the reasons set forth above, the rejections of Claims 1-37 are improper as the Examiner has failed to establish a prima facie case of obviousness under 35 USC §103.

Respectfully submitted,
Shell S. Simpson

By ___/Jack H. McKinney/_____
    Jack H. McKinney
    Reg. No. 45,685

September 12, 2007

APPENDIX OF CLAIMS INVOLVED IN THE APPEAL

1.   (Previously presented) A method for monitoring a web-based service, comprising the steps of:

receiving automatically at a client a service reference to a status of a job in a network service;

adding the service reference to a bookmark list on the client; and

removing automatically the service reference from the bookmark list on the client when the job is completed by the network service.

2.   (Previously presented) The method as defined in claim 1, wherein the network service is a printer service, and the job is a print job.

3.   (Previously presented) The method as defined in claim 1, wherein the service reference is a URL to a status page for the network service.

4.   (Previously presented) The method as defined in claim 1, wherein the service reference is provided when the network service is accessed.

5.   (Previously presented) A method for monitoring a web-based service, comprising the steps of:

receiving automatically in a user's personal imaging repository in an autonomous network service a service reference to a status of a job in a job-performing network service, wherein the autonomous network service is independent from the job-performing network service and does not facilitate performance of the job at the job-performing network service;

adding the service reference to a bookmark list in the user's personal imaging repository; and

removing automatically the service reference from the bookmark list in the user's personal imaging repository when the job is completed by the job-performing network service.

6.   (Original) The method as defined in claim 5, wherein the bookmark list is maintained within a user profile in the user's personal imaging repository.

7.   (Previously presented) The method as defined in claim 5, further comprising the step of determining the status of the job.

8.   (Previously presented) The method as defined in claim 7, wherein the determining step comprises querying the network service to determine if a job impediment has occurred.

9   (Previously presented) The method as defined in claim 7, wherein the determining step comprises querying the network service to determine an indication of what amount of the job is complete.

10.   (Previously presented) The method as defined in claim 7, wherein the determining step comprises receiving a message from the network service indicating an event.

11.   (Original) The method as defined in claim 10, wherein the receiving a message step comprises receiving the message at a bookmark management software.

12.   (Original) The method as defined in claim 10, wherein the receiving a message step comprises receiving a message of one or more of the following: print job completed, copies printed, and error.

13.　(Previously presented) The method as defined in claim 5, further comprising storing the service reference to a storage associated with the user.

14.　(Previously presented) A method for monitoring a web-based service, comprising the steps of:

receiving automatically at a client a service reference to a status of a job in a network service;

adding the service reference to a bookmark list on the client; and

removing automatically the service reference from the bookmark list on the client when the job is completed by the network service

further comprising the step of adding a clickable reference to a user screen on the client that is associated with the service reference to access a status web page displaying direct or indirect information about the status of at least one job.

15.　(Original) The method as defined in claim 14, further comprising the step of adding a plurality of clickable references to be displayed on a user screen, each clickable reference associated with a different service reference for opening a different status web page having information about the status of its respective job.

16.　(Previously presented) The method as defined in claim 14, further comprising the step of adding a window associated with the network service to a user screen to display therein a status web page with direct or indirect information about the status of at least one job.

17.　(Previously presented) The method as defined in claim 14, wherein the bookmark list lists only job status service references.

18.　(Previously presented) A method for monitoring a web-based service, comprising the steps of:

receiving automatically at a client a service reference to a status of a job in a network service;

adding the service reference to a bookmark list on the client;

determining the status of the job; and

removing automatically the service reference from the bookmark list on the client when the job is completed by the network service, wherein the determining step comprises receiving a message from the service indicating an event; and

wherein the removing step includes the step of automatically removing the service reference if no message indicating an event is received from the network service for a predetermined period of time.

19.   (Previously presented) A system for monitoring a web-based service, comprising:

a component for receiving automatically at a client a service reference to a status of a job in a network service;

a component for adding the service reference to a bookmark list on the client; and

a component for removing automatically the service reference from the bookmark list on the client when the job is completed by the network service.

20.   (Previously presented) A program product for monitoring a web-based service, comprising machine readable program code for causing a machine to perform the following method steps:

receiving a service reference automatically at a client to a status of a job in a network service;

adding the service reference to a bookmark list on the client; and

removing automatically the service reference from the bookmark list on the client when the job is completed by the network service.

21.   (Previously presented) The program product as defined in claim 20, wherein the network service is a printer service, and the job is a print job.

22.   (Previously presented) The program product as defined in claim 20, wherein the service reference is a URL to a status page for the network service.

23.   (Previously presented) The program product as defined in claim 20, wherein the service reference is provided when the network service is accessed.

24.   (Previously presented) A program product for monitoring a web-based service, comprising:

a machine-readable medium that includes disposed thereon computer readable program code, that when executed, causes the following steps to be performed:

receiving automatically in a user's personal imaging repository in an autonomous network service a service reference to a status of a job in a job-performing network service, wherein the autonomous network service is independent from the job-performing network service and does not facilitate performance of the job at the job-performing network service;

adding the service reference to a bookmark list in the user's personal imaging repository; and

removing automatically the service reference from the bookmark list in the user's personal imaging repository when the job is completed by the job-performing network service.

25.   (Previously presented) The program product as defined in claim 24, wherein the bookmark list is maintained within a user profile in the user's personal imaging repository.

26.   (Previously presented) The program product as defined in claim 24, further comprising program code for the step of determining the status of the job.

27.   (Previously presented) The program product as defined in claim 26, wherein the determining step comprises querying the network service to determine if a job impediment has occurred.

28.   (Previously presented) The program product as defined in claim 26, wherein the determining step comprises querying the network service to determine an indication of what amount of the job is complete.

29.   (Previously presented) The program product as defined in claim 26, wherein the determining step comprises receiving a message from the network service indicating an event.

30.   (Previously presented) The program product as defined in claim 29, wherein the receiving a message step comprises receiving the message at a bookmark management software.

31.   (Previously presented) The program product as defined in claim 29, wherein the receiving a message step comprises receiving a message of one or more of the following: print job completed, copies printed, and error.

32.   (Previously presented) The program product as defined in claim 24, further comprising program code for storing the service reference to a storage associated with the user.

33.   (Previously presented) A program product for monitoring a web-based service, comprising:

a machine-readable medium that includes disposed thereon computer readable program code, that when executed, causes the following steps to be performed:

receiving automatically at a client a service reference to a status of a job in a network service;

adding the service reference to a bookmark list on the client; and

removing automatically the service reference from the bookmark list on the client when the job is completed by the network service,

further comprising the step of adding a clickable reference to a user screen on the client that is associated with the service reference to access a status web page displaying direct or indirect information about the status of at least one job.

34.   (Previously presented) The program product as defined in claim 33, further comprising program code for the step of adding a plurality of clickable references to be displayed on a user screen, each clickable reference associated with a different service reference for opening a different status web page having information about the status of its respective job.

35.   (Previously presented) The program product as defined in claim 33, further comprising program code for the step of adding a window associated with the network service to a user screen to display therein a status web page with direct or indirect information about the status of at least one job.

36.   (Previously presented) The program product as defined in claim 33, wherein the bookmark list lists only job status service references.

37.   (Previously presented) A program product for monitoring a web-based service, comprising:

a machine-readable medium that includes disposed thereon computer readable program code, that when executed, causes the following steps to be performed:

receiving automatically at a client a service reference to a status of a job in a network service;

adding the service reference to a bookmark list on the client;

determining the status of the job by receiving a message from the network service indicating an event; and

removing automatically the service reference from the bookmark list on the client when the job is completed by the network service or if no message indicating an event is received from the network service for a predetermined period of time.